

# ToM-Dyna-Q: on the integration of reinforcement learning and machine Theory of Mind

Dan Kröhling, Ernesto Martínez

INGAR (CONICET/UTN)  
Avellaneda 3657  
Santa Fe, Argentina

**Abstract.** The capacity to understand others, or to reason about others' ways of reasoning about others (including us), is fundamental for an agent to survive in a multi-agent uncertain environment. This reasoning ability, commonly known as Theory of Mind, is instrumental for making effective predictions over others' future actions and learning from both real and simulated experience. In this work, a novel architecture for model-based reinforcement learning in a multi-agent setting is proposed. The proposed architecture, called ToM-Dyna-Q, integrates ToM simulation alongside with the well-known Dyna-Q architecture to account for artificial cognition in a shared environment inhabited by multiple agents interacting with each other. Results obtained for the two-player competitive game of Tic-Tac-Toe demonstrate the importance for a given agent of learning, reasoning and planning based on mental simulation modeling of other agents' goals, beliefs and intentions.

**Keywords:** Intelligent agents, prediction machines, reinforcement learning, Theory of Mind.

## 1 Introduction

From the beginning of mankind, we humans have experienced that it is practically impossible to survive without successfully competing and cooperating with others within a shared environment. The need to collaborate strategically with others has lead to the emergence of complex markets, where each individual must interact with his pairs in order to accomplish his own goals. Through time, this interaction contributed to the arise of increasingly sophisticated social skills that required humans to reason about the world they inhabit, what knowledge and understanding others have, what do they think we know they know and think, and so forth. This particular recursive type of reasoning about others has lately gained attention, and is usually referred to as Theory of Mind (ToM).

Theory of Mind denotes the ability to represent the mental states of others, including their beliefs, intentions and goals [2, 13], which can be used advantageously over shallower ways of reasoning [12]. Scientists have wondered if this

ability may be present not only in humans and other few species of animals [6], but also implemented by machines and rational artificial agents. Actually, it could [10, 11], and a number of recent research efforts have proven ToM effectiveness [3, 4, 14, 19].

As stated in [9], a great variety of methods try to address implementation details of ToM-based artificial agents [3, 8, 19] and that also happens with some learning agents in general [5, 7]. However, to the best of our knowledge, a conceptual architecture for tightly integrating reinforcement learning in a multi-agent setting with simulating ToM reasoning for planning and deliberation is lacking.

Bearing in mind model-based Q-learning in multi-agent systems, the novel ToM-Dyna-Q architecture is proposed. The Q-learning [18] algorithm is used for policy learning based on both actual and simulated experience. However, other reinforcement learning algorithms such as *Sarsa* or Bayesian Q-learning can be used. Q-learning will suffice to keep things simple enough and help us concentrate on answering the questions that drive our computational study: is ToM useful to gain a competitive advantage over other learning agents? Does ToM accelerate the learning curve of more sophisticated reasoning agents? Is an agent's best strategy to play always optimally?

To address the foregoing questions, the simple Tic-Tac-Toe game will be used. A group of agents was created (named Q, R, S, and T), some of them with the opportunity to learn about the game and their opponents, and some of them without that capability. We then provide certain agents the possibility to use different levels of ToM reasoning about the opponent, to test intuitions regarding the role to ToM-based learning in artificial agents. Finally, we set them against each other in tournaments of Tic-Tac-Toe games.

To begin with, a short introduction to previous works on Theory of Mind and model-based Q-learning is given in Section 2. Then, in Section 3, the proposed architecture for a ToM-Dyna-Q agent (Jack) is presented. In Section 4, a number of computational experiments made are discussed together with results obtained for different scenarios. Finally, some concluding remarks about our research efforts in implementing ToM for multi-agent learning are made in Section 5.

## 2 Previous work

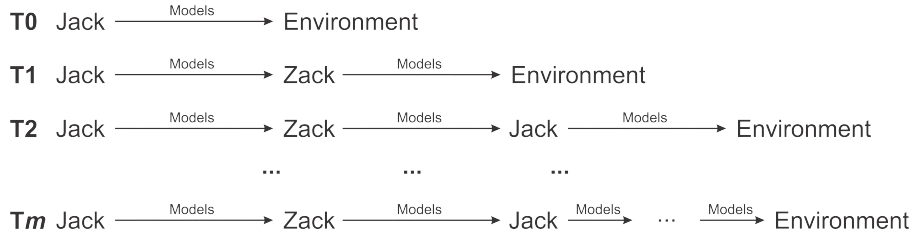
### 2.1 Theory of Mind

Following the definition given by Singer et al. [13], it can be said that Theory of Mind, or the equivalent term “mentalizing”, connotes an agent's ability to cognitively represent the mental states of others, including intentions, knowledge, goals, beliefs, and models. As mentioned earlier, this is not only a human capacity, but also is present in a few animal species and now intelligent artificial agents too [10, 11].

In previous works, it has been proved that deception and manipulation could provide agents a social competitive advantage [4]. Thus, we could state that

ToM, that is, the capacity to infer other agent's thoughts and states as well as effectively predicting their next actions [1], is a natural consequence of learning and deliberation in rational agents in order to effectively interact with others. In Section 4, this ability to understand others (or to reason about other agents' reasoning) in strategic interactions between two players is analyzed. A previous illuminating study could be found also in [17].

Fig. 1 will serve us to explain the concept of ToM. Suppose there are two agents, Jack (our focal agent) and Zack, interacting in a certain environment. As a ToM-0 agent (or T0), Jack makes a model out of the environment but does not take into account any other agent (Zack, for instance) in it. In other words, Jack only sees the effect of Zack's actions over the environment and himself, but could not identify Zack as an entity capable of responding to his actions and acting purposefully. As a T1 agent, Jack models not only the environment as a set of interacting passive objects, but also distinguishes the presence of others as T0 agents. That is, Jack interprets that a Zack thinks as he would do if the environment were agent-free. As could be imagined, a T2 Jack would model a Zack as if he were a T1 agent, and so it continues. In general, a Tm Jack would model a Zack as a Tm - 1 agent.



**Fig. 1.** Theory of Mind explained.

## 2.2 Dyna-Q (model-based Q-learning)

Q-learning [18] is a model-free algorithm created for learning from a sequence of reinforcements and state transitions, which consists of a function that iterates over the expected cumulative rewards for future time steps in episodic interactions between an agent and its environment. We will explain it succinctly: given a state  $s_t$  observed by a Q-learning agent in a certain environment, this algorithm determines the next action  $a_t$  from all the agent's possible actions  $A$  she can perform following a policy  $Q(s, a)$ . Then, the environment returns both a reward  $r$ , that will be used as a reinforcement signal or hint, and its next state  $s_{t+1}$  to the agent. The algorithm takes this information and actualizes the policy  $Q(s, a)$  at the end of each episode according to the learning rule:

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha [r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (1)$$

In equation 1 there are two hyper-parameters used to influence the learning curve and the long-term view of action effects on Q-values. Firstly, the learning rate  $\alpha$  measures the importance given to prediction errors in Q-values updates. Secondly, the discount rate  $\gamma$  establishes how much our agent will look up into the future for planning and learning. As a common rule of thumb, these parameters are usually set to 0,1 and 0,9, respectively [15]. There is a third parameter,  $\epsilon$ , that is used to determine the balance between exploration and exploitation of the learning agent based on what she knows and its uncertainty. When  $\epsilon$  is set near to 0, the algorithm will exploit what it already knows for certain, not exploring for alternative courses of action, even if there may exist better ones. When  $\epsilon$  is set near to 1, the algorithm will explore the state space as much as it can to reduce uncertainty, but would not consider those rewarding courses of action already found, despite how good they are.

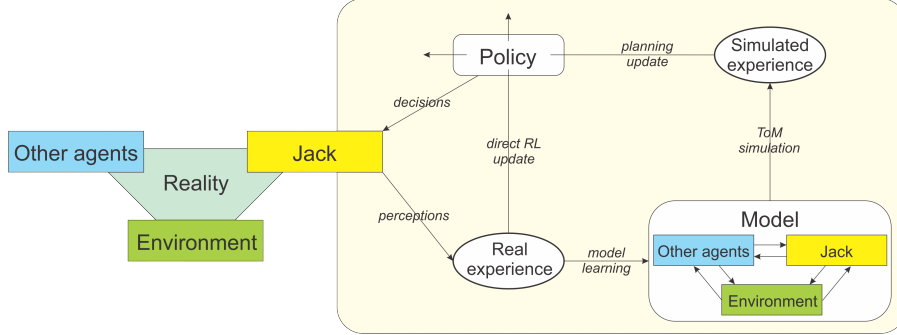
Dyna-Q [15], or model-based Q-learning, is an extension to the mentioned Q-learning algorithm, that incorporates a model of the environment to combine real and simulated experience in Q-values updates. Dyna-Q uses a generative model of the environment and allows the Q-learning agent to plan in advance his actions based on simulating sequences of action-state-rewards. The main assumption behind the generative model is that it does not change in response to the agent's learning process. In Dyna-Q, the agent learns not only its policy but also improves the model of its environment. The Dyna-Q does not apply to a multi-agent setting because, as one agent learns, the generative model also changes due to the response of other agents in its environment. An extension to Dyna-Q is thus needed to account for the effect of one agent learning, reasoning and planning on the other agents in the environment.

### 3 ToM-Dyna-Q

In this section, we propose to overhaul completely the Dyna-Q architecture so that a learning agent such as Jack could not only model the perceivable changes (the state transitions) in the environment due to its actions, but also to identify and mentally represent other agents reasoning and learning processes. The resulting model-based learning architecture, aptly named ToM-Dyna-Q, makes room for a given agent Jack to elaborate and plan about the knowledge it has over other agents' knowledge, policies, models, goals, etc., including knowledge and models he believes the others think Jack has about them. This novel multi-agent learning and planning framework makes room for creating ToM learning agents with different levels of sophistication. In Fig. 2<sup>1</sup>, the main building blocks of the ToM-Dyna-Q learning in a multi-agent environment are shown.

Reinforcement learning using ToM-Dyna-Q allows for a more comprehensive framework for learning a model of an agent's real world (or his reality) through reasoning about other agents and environmental objects separately. Real experience for the agent is now highly informative about other agents' models, beliefs

<sup>1</sup> The original figure was taken from [15] and conveniently adapted.



**Fig. 2.** Tom-Dyna-Q for model-based learning and reasoning in multi-agent systems.

and strategies. Based on ToM simulations, a given agent learns adaptations to his policy which, when implemented in the real world, gives rise to perceptions upon which direct reinforcement learning (RL) is practiced. Thus, by planning and reasoning using a gradually refined model of the multi-agent environment, the policy used while interacting with the actual environment probes other agents responses. When compared with predictions about expected actions, a signal error allows for further elaborations and refinements.

## 4 Computational experiments

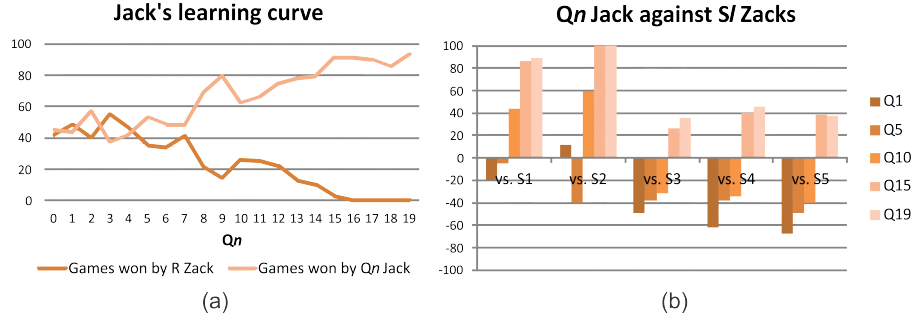
### 4.1 Setting

In this section, the setting for the computational experiments is detailed. In order to show the advantages that could be obtained from ToM-Dyna-Q, the English version of the Tic-Tac-Toe is used as a representative example. Two agents, Jack and Zack (in the different configurations discussed below), will play against each other in sessions of 100 games to assess about the competitive edge of “mentalizing” the opponent.

The configurations Jack and Zack may adopt are:

- **Q-configuration.** In this configuration, an agent would have learned to play Tic-Tac-Toe using the well-known Q-learning algorithm playing against another instance of himself. Nineteen experience levels were created corresponding to the different number of games that the agent had played, with the 19th level corresponding to a total of 10 million episodes played. In general, we will name  $Q_n$  an agent of an  $n$  level of experience.
- **R-configuration.** An R agent would be an agent that plays completely at random, regardless its opponent’s mental state, expertise, or board state.
- **S-configuration.** Five heuristic strategies typically used by human players for Tic-Tac-Toe (e.g., as a start choice, to play always in the board center). From the simplest to the most complicated one, they will be called as  $S_l$ ,  $l$  being their heuristic complexity level.

In Fig. 3, it is shown the different  $Qn$  levels obtained for an agent, playing against R and S agents, to highlight later to what extent ToM-based learning may improve the cumulative utilities the agents get.

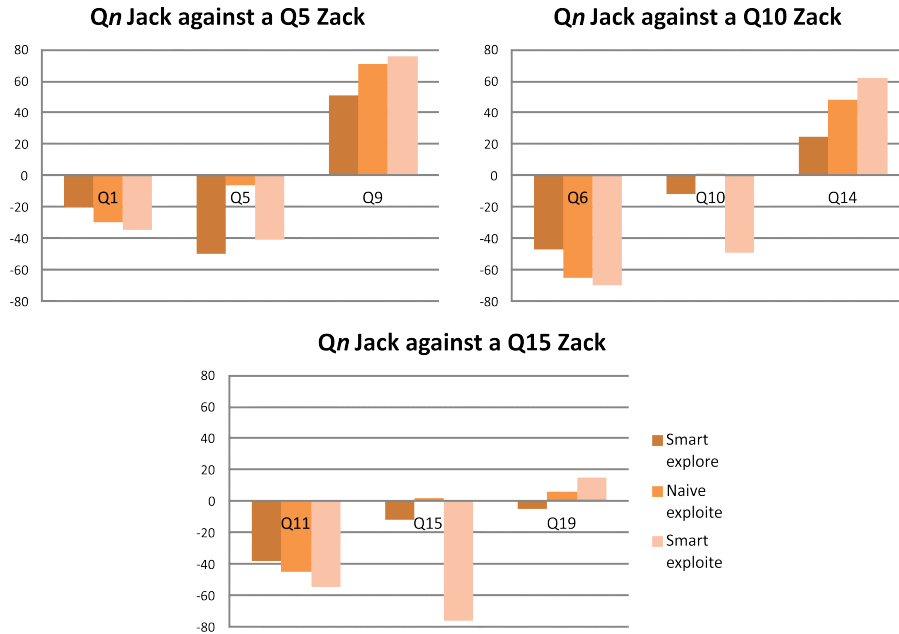


**Fig. 3.** (a) Amount of games won by a  $Qn$  Jack and a R Zack playing against each other in tournaments of 100 games, with Jack's  $Qn$  levels varying from 0 to 19. (b) Rewards obtained by five different  $Qn$  Jacks against five different S/Zacks.

- **T-configuration.** ToM agents are identified by  $Tm$ , where  $m$  is the level of Theory of Mind our agent is going to use, as described earlier in Fig. 1. In this configuration, a  $Tm$   $Qn$  agent is one such that it could play as a  $Qn$  agent, but may also behave as an agent with lower experience levels  $[1; n - 1]$  in order to mislead his opponents or resort to a higher  $\epsilon$  in order to explore more and learn over his opponent knowledge and expertise. The ToM agent will model his opponent as explained later, and then will consider this information to be the state over which he will reason, plan and learn.

The rationale for ToM agents when modeling their opponents is based on the number of games won by a given opponent considering the last ten games. If an opponent has won ten out of the last ten, then he is certainly a tough opponent. Accordingly, the ToM agent will try to learn as much as it can from such an opponent. On the contrary, If the opponent has won zero, then it is easy to beat, and a ToM agent will try to mislead such an easy opponent to slow down its learning process. The demonstration of this seemingly intuitive reasoning is presented in Fig 4. The rewards gathered by any of our agents are based on the difference between the number of games won and the number of games lost, leaving aside the games that end up in a draw.

As can be seen, when the  $Qn$  level of Jack is lower than that of Zack, then the better choice for Jack is to explore more by increasing his  $\epsilon$ . When the  $Qn$  level of Jack is almost equal than that of Zack, then for Jack is better just playing as he knows best, in order to win more games. Finally, when the  $Qn$  level of Jack is greater than that of Zack, then the better choice for him is to exploit his knowledge, but lowering the ToM level of his strategy, that is to say, picking



**Fig. 4.** The intuition that uses our agent to play given the experience he thinks his opponent has. In the vertical axis are presented Jack's rewards when playing against a Q5, a Q10, and a Q15 Zack in tournaments of 100 games.

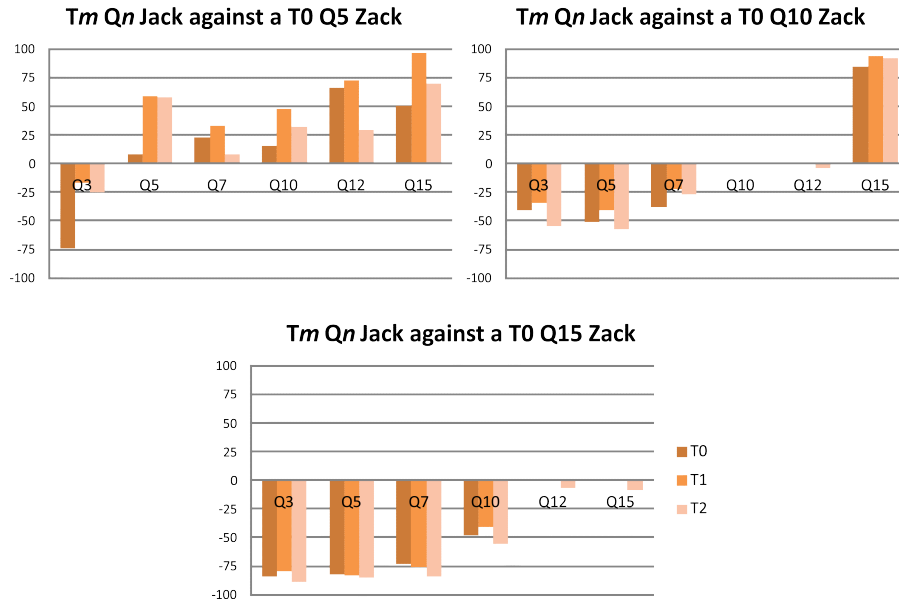
consciously over his  $[1;n]$  available levels in order to prevent the opponent to learn from his own expertise.

## 4.2 Results and analysis

In this section, results obtained when a  $T_m$  Jack is set to play against a  $T_0$  Zack are presented. As was previously mentioned, these agents are engaged in playing tournaments of 100 games.

In Fig. 5, the rewards obtained by the agent Jack when playing with different levels of Theory of Mind against a  $T_0$  agent are shown. It is noteworthy that, whichever his  $Q_n$  level of experience is, his best choice is always to play as a  $T_1$  agent. This is not a surprise, given that a  $T_0$  and a  $T_2$  agent construct models that do not correspond to the actual reality, and thus fail to predict Zack's playing strategy.

The same situation could be seen the next two graphics. However, the differences blur a little when the opponent's experience level is significantly high. Such a result may be attributed to the fact that, when Jack's opponent strategy is actually very hard to defeat, it cannot establish a policy sufficiently good to cope with such a rival. On the other hand, 100 games may be too small a sample of games, which does not allow Jack to gain sufficient knowledge to successfully be-



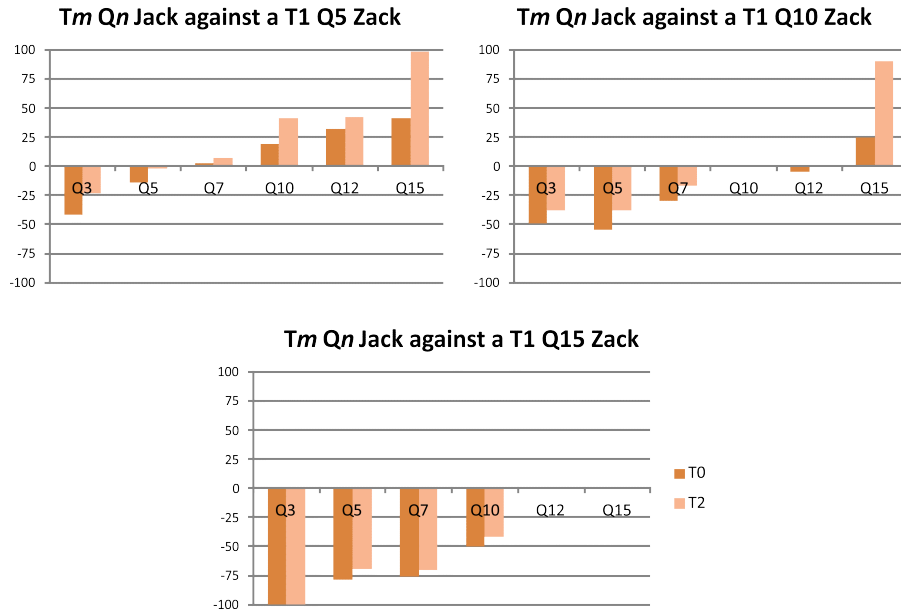
**Fig. 5.** In the vertical axis are presented Jack's rewards when playing with different levels of sophistication against a T0 Zack with Q5, Q10, and Q15 experience levels.

have in the game. For instance, a Q5 agent has only played two hundred learning episodes in Tic-Tac-Toe before the tournament, visiting only 156 possible states in the board, whereas a Q15 agent has experienced 500000 learning episodes, visiting as many as 3947 of a total of 8744 relevant states in the board, after the elimination of some symmetrical board states.

In Fig. 6 could be seen that a T2 Jack makes the best out of him when he is playing against a T1 Zack, in contrast to what happened in the T0 Zack scenarios. In this case, the model he makes of his opponent is accurate, and allows him to pick the right actions and get better results.

As a summary, it can be stated that, for a given agent, the use of ToM for planning and learning could give a competitive advantage over other learning agents, but should not be overestimated. If an agent fails to identify the depth of the reasoning his opponents resort to, then it may fool itself by taking suboptimal decisions based on too complex reasoning. Even worse, actions taken may be far away from those reasonable courses of action the agent may learn without modeling its opponent at all. The correct use of ToM does accelerate learning, while identifying the proper actions to take (to exploit or explore) based on the experience the opponent has in the environment. Finally, we could say that the best strategy of an agent is not always to play the one he knows is his best play, but to identify the goals, intentions and models other agents have about the environment and agents within it, as well as and the beliefs they think it has about the them, their intentions, goals and strategies.





**Fig. 6.** In the vertical axis are presented Jack's rewards when playing with T0 and T2 against a T1 Zack.

## 5 Concluding remarks

In this work, a novel architecture for model-based reinforcement learning in a multi-agent environment is proposed. The novel architecture ToM-Dyna-Q may integrate different learning rules and models for accounting about other agents reasoning, planning and learning, including Bayesian Q-learning, actor-critic, eligibility traces [16], etc. Many applications for the Internet of Things could be approached by the proposed architecture. Automated negotiations, autonomous driving, and emergent scheduling, were the main drivers for our proposal.

The English version of the Tic-Tac-Toe game was presented as a toy example, where a proper integration of Theory of Mind with reinforcement learning show promising results. Through a number of experiments with different types of agents proposed, our research questions were addressed, demonstrating that the correct use of ToM by an agent may provide him a competitive advantage over other learning agents that do not reason and plan bearing in mind that other agents may also have a mind on their own.

## References

1. A. Agrawal, J. Gans, and A. Goldfarb. *Prediction machines: The simple economics of artificial intelligence*. Harvard Business Review Press, Boston Massachusetts, 2018.

2. C. L. Baker, J. Jara-Ettinger, R. Saxe, and J. B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4):598, 2017.
3. S. de Jong, D. Hennes, K. Tuyls, and Y. Gal. Meta-strategies in the colored trails game. *Belgian/Netherlands Artificial Intelligence Conference*, (c):2–6, 2011.
4. H. de Weerd, R. Verbrugge, and B. Verheij. How much does it help to know what she knows you know? an agent-based simulation study. *Artificial Intelligence*, 199–200:67–92, 2013.
5. R. Dearden, N. Friedman, and S. Russell. Bayesian q-learning. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, AAAI '98/IAAI '98, pages 761–768, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.
6. J. Joiner, M. Piva, C. Turrin, and S. W. C. Chang. Social learning through prediction error in the brain. *npj Science of Learning*, 2(1):8, 2017.
7. D. S. Leslie and E. J. Collins. Individual q-learning in normal form games. *44(2):495–514*, 2005.
8. J. Pöppel and S. Kopp. Satisficing models of bayesian theory of mind for explaining behavior of differently uncertain agents. *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018)*, Stockholm, Sweden., (July), 2018.
9. S. Qi and S.-C. Zhu. Intent-aware multi-agent reinforcement learning. *CoRR*, abs/1803.02018, 2018.
10. N. C. Rabinowitz, F. Perbet, H. F. Song, C. Zhang, S. M. A. Eslami, and M. Botvinick. Machine Theory of Mind. *ArXiv e-prints*, Feb. 2018.
11. B. T. Revell. Deepmind ai is learning to understand the 'thoughts' of others. *New Scientist*, (February):1–5, 2018.
12. I. Sher, M. Koenig, and A. Rustichini. Children's strategic theory of mind. *Proceedings of the National Academy of Sciences*, 111(37):13307–13312, 2014.
13. T. Singer and A. Tusche. Understanding others: Brain mechanisms of theory of mind and empathy. *Neuroeconomics: Decision Making and the Brain*, pages 513–534, 2014.
14. C. A. Stevens, N. A. Taatgen, and F. Cnossen. Instance-based models of metacognition in the prisoner's dilemma. *Topics in Cognitive Science*, 8(1):322–334, 2016.
15. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. Adaptive computation and machine learning series. The MIT Press, Cambridge MA, second edition edition, 2018.
16. R. S. Sutton, D. Precup, and S. Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1-2):181–211, 1999.
17. F. B. von der Osten, M. Kirley, and T. Miller. The minds of many: Opponent modelling in a stochastic game. *IJCAI International Joint Conference on Artificial Intelligence*, pages 3845–3851, 2017.
18. C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK, 1989.
19. W. Yoshida, R. J. Dolan, and K. J. Friston. Game theory of mind. *PLoS Computational Biology*, 4(12), 2008.